# Adaptive Thresholding Techniques

EECS5330
Image Analysis and Computer Vision
Edris Amin Kris Armstrong

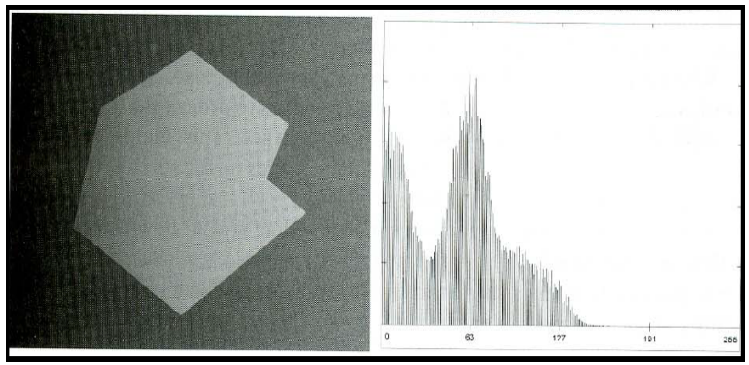
# Overview

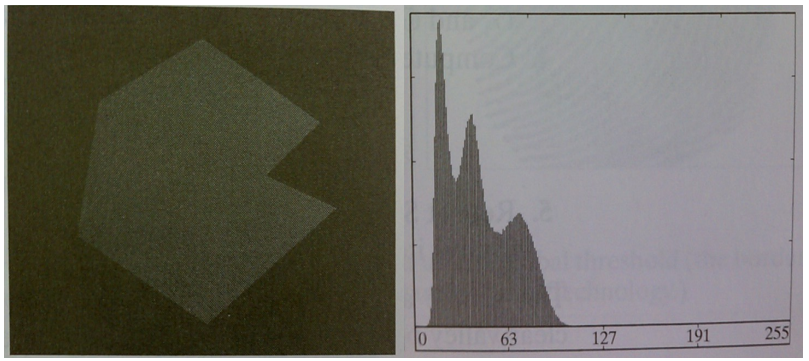
Image Binarization

- Global Thresholding
  - applied to the entire image
  - global thresholding has difficulty with images that have:
    - changes in illumination
    - scanning errrors
    - low resolution
    - complex documents(mixed text and graphics)
- Variable Threholding
  - any method that uses more than one threshold for the image
- Adaptive Thresholding
  - uses a local neighborhood to calculate the threshold for each pixel

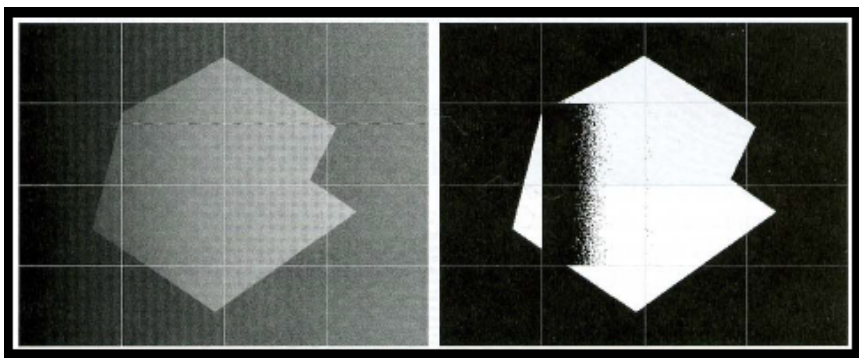
# Binarization Using Thresholding

- Basics
  - Global thresholding.




  - Adaptive thresholding



# Adaptive Thresholding

- Otsu's Algorithm
  - Select Threshold level T
  - 2 Sets of pixels $S_o$, $S_b$
  - Weight & variance
    object & background
  - Variances: within-set
    between-set, total
  - Select Threshold for maximum
    between-set vs within-set ratio

$$W_o = \sum_{i=1}^{T} p(i);\ W_b = \sum_{i=T+1}^{L} p(i);\ W_b = 1 - W_o$$

$$\mu_0 = \sum_{i=1}^{T} \frac{ip(i)}{W_0};\ \mu_b = \sum_{i=T+1}^{L} \frac{ip(i)}{W_b};\ \mu_T = W_0\mu_0 + W_b\mu_b$$

$$\sigma_o^2 = \sum_{i=1}^{T} \frac{(1-\mu_o)^2 p(i)}{Wo};\ \sigma_b^2 = \sum_{i=T+1}^{L} \frac{(1-\mu_b)^2 p(i)}{Wb}$$

$$\sigma_W^2 = \omega_0\sigma_0 + \omega_1\sigma_1 \qquad \sigma_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2$$

$$\sigma_T^2 = \sum_{i=1}^{L} (i-\mu_T)^2 p(i) \qquad \eta = \max_{1 \le T \le L} \frac{\sigma_B^2}{\sigma_W^2}.$$

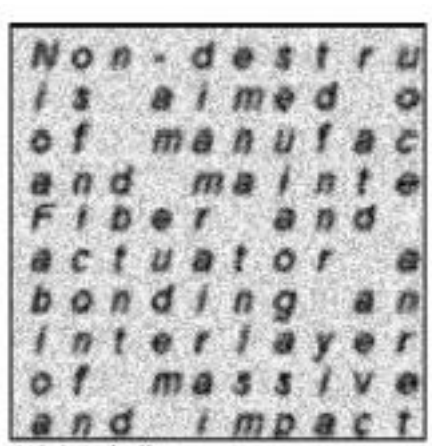
- Niblack's Algorithm
  - Choose a square scanner
    - size: b x b
  - Apply thresholding within square
    -

$$m(x,y) = \frac{1}{b^2}\sum_j \sum_i f(i,j) \qquad \sigma^2(x,y) = \frac{1}{b^2}\sum_j \sum_i (m(x,y) - f(i,j))^2$$

$$T(x,y) = m(x,y) + k * \sigma^2(x,y)$$

# Adaptive Thresholding

- **Bernsen's Method**
  - A classic method that others use for comparison
  - Finds threshold of the center pixel, T(i,j), of a n x m set of pixels by calculating the mean of the maximum and the minimum value of the pixels in the window
  - T(i,j) = 0.5 * {max[c(i + x, j + y)] + min[c(i + x, j + y)} where x,y iterate over all pixels in the window
  - b(x,y) = 1 if [c(i,j) - T(i,j)] > 0
    0 otherwise



Arial,14,italic

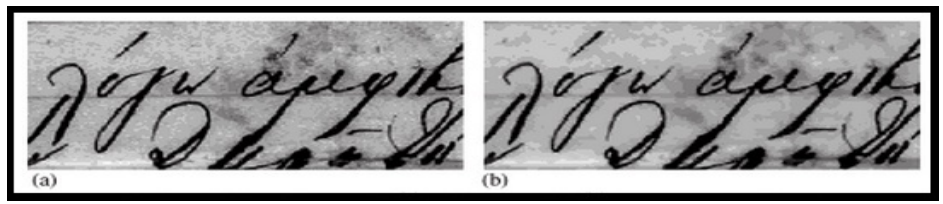
Local_Bernsen, S=0.078

# Adaptive Thresholding

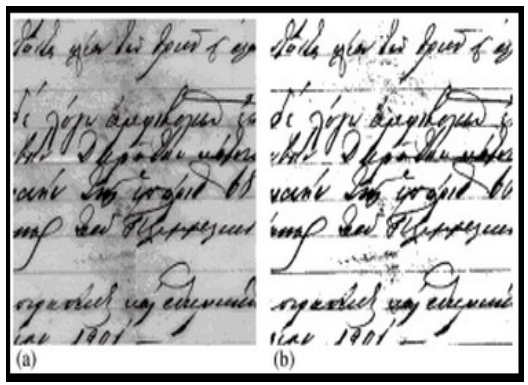
- **Sauvola Method**
  - a hybrid method that adapts the threshold window to either text or graphic types.
  - divide the image into small windows and use TD to determine the type of threshold to use
  - TD = [2P(i,j)-[P(i-1,j)+P(i,j-1)]  over every p(i,j)
  - For graphics use:
    - calculate a weighting system based on local histograms
    - use weights to determine the threshold for each pixel
  - For text use: (modified Niblack)
    - T(i,j) = m(i,j) + {1 + k[var(i,j) / (R - 1)]}
      - R = the dynamic range of the standard deviation

# Adaptive Thresholding

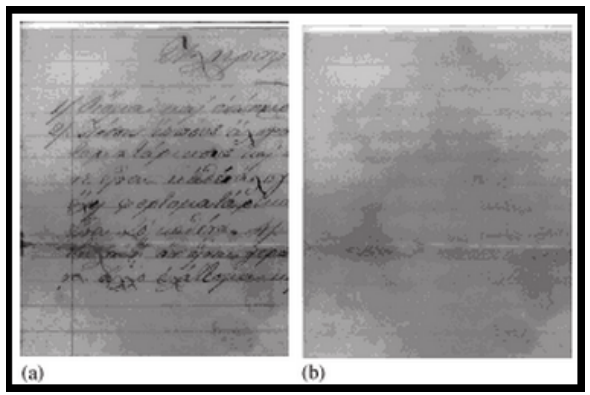
- Gato's Method
  - Wiener applied
    Smoothing background




  - Binary image using
    Sauvola's alogrithm




  - Background image
    select threshold d



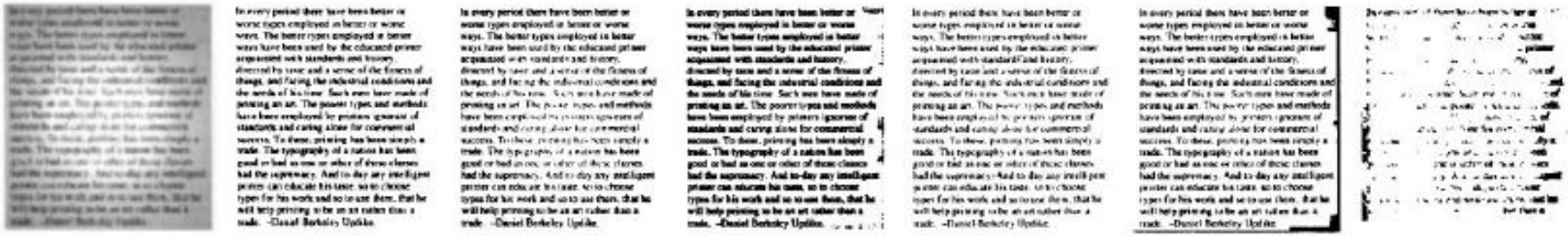

  - Apply Thresholding
    - T(i,j) = 1 for { backround(i,j) - image(i,j) } > d

# Adaptive Thresholding

## Comparison of Techniques



**Result images and ranked benchmarking evaluation results**




| | |
|---|---|
| Sauvola | 95.8 % |
| Niblack | 89.9 % |
| Eikvil | 82.1 % |
| Bernsen | 78.6 % |
| Parker | 38.4 % |

**Description of the BM event:**

-15 textual document images with various fonts and typefaces.
-Each document having implanted illumination.
-The degree of illumination is varied (non-illuminated=>bad illumination).
-The above visual results show 20% illumination factor applied to GT image map.

Fig. 12. Visual and numeric results on the comparison algorithms applied to illuminated, textual images.

# A New Method

[Yu-Ting Pai, et. al.]

- Problems with current Adaptive Thresholding methods
  - Recursive → Time + Memory + Computation
- Fusion
  - Smoothing
  - Select box size
  - Segment Image
  - Threshold segments

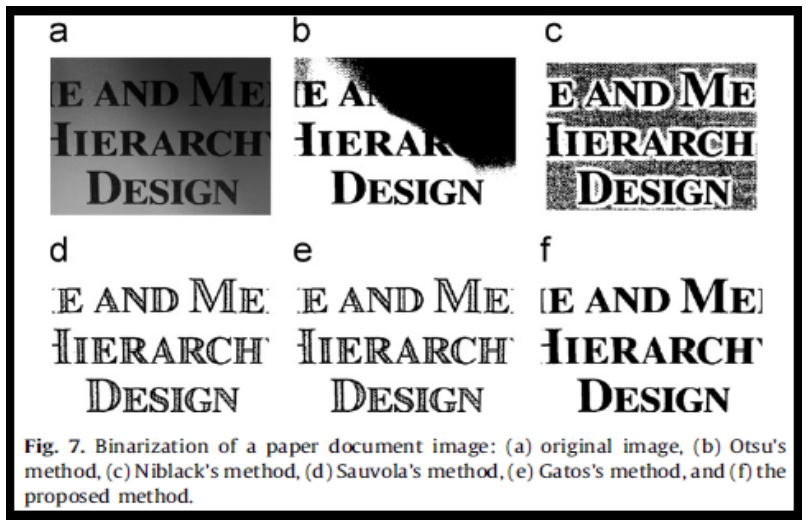



Fig. 7. Binarization of a paper document image: (a) original image, (b) Otsu's method, (c) Niblack's method, (d) Sauvola's method, (e) Gatos's method, and (f) the proposed method.

  - Computational Complexity
    - M x N size, L levels, b block size

Otsu: $MN+7L^2+5L-12$ Niblack: $(4b^2+3)MN$

Fusion: $N(M-1)$

# References


B. Gatos, I. Pratikakis, S.J. Perantonis
"Adaptive degraded document image binarization"
Sept. 2005 The Journal of the Pattern Recognition Society

Yu-Ting Pai, Yi-Fan Chang, Shanq-Jang Ruan
"Adaptive thresholding algorithm: Efficient computation technique based on intelligent block detection for degraded document images"
2010 The Journal of the Pattern Recognition Society

R.C. Gonzales, R.E. Woods
*Digital Image Processing Third Edition*
2008 Prentice Hall

# References

J. Brensen "Dynamic Thresholding of Gray Level Images," ICPR'86 Proc. Intl. Patt. Recog., pp 1251-1255 (1986)

J. Sauvalo and M. Pietaksinen, "Adaptive Document Image Binarization," Pattern Recog., 33 pp.225 - 236 (2000)

N. B. Venkateswarluh and R.D. Boyle, "New segmentation techniques for document images," Image Vis. comput. 13 pp. 573 - 583 (1995)